

UAB Bus: Aplicación móvil para usuarios de los buses internos de la UAB

Sergi Luque Perfecto

Resumen– Actualmente, es habitual que cada localidad tenga una aplicación para smartphone para que sus usuarios puedan consultar información sobre el servicio de su red de transporte público. Pero muchas veces estas no cumplen las necesidades de muchos usuarios. La aplicación web de buses de la UAB se ha quedado atrás en funcionalidades respecto a aplicaciones de otras localidades. Por eso, en este proyecto se ha creado una aplicación alternativa a la aplicación actual para añadir nuevas características.

Palabras clave– Flutter, Bus, Firebase, Aplicación móvil multiplataforma, Android, Transporte público

Abstract– Nowadays, it's common for every locality to have its own smartphone application to consult the information of the public transport service. However, they often do not meet the needs of many users. The UAB bus web application has lagged behind in functionality compared to applications from other locations. Therefore, in this project an alternative application to the current application has been created to add new features.

Keywords– Flutter, Bus, Firebase, Cross-platform mobile application, Android, Public transport

1 INTRODUCCIÓN

LA UAB tiene una red de buses interno de uso gratuito para facilitar la movilidad dentro del campus Bellaterra. Esta se compone por estas 6 líneas:

- L1 Renfe - Rectorat (per Ciències)
- L2 Renfe - Vila (per Medicina)
- L3 Ciències i Biociències - Eureka
- L3 Eureka - Ciències i Biociències
- L4 Renfe - Rectorat (per Eix Central)
- L5 Renfe - Vila (per Eix Central)

Para saber el horario de los buses que pasan por una parada se puede consultar en el panel informativo de la misma o desde la web oficial [1]. La mayoría de usuarios utilizan la primera opción para consultarlo por su facilidad y rapidez. Eso hace que cuando el horario del panel se encuentre dañado se tenga que consultar a través de la web con sus costes en tiempo y uso de datos móviles, como se puede ver en la Figura 1.

La página web actual tiene una versión adaptada para pantallas de ordenador y otra para pantallas de smartpho-

- E-mail de contacto: sergi.luque@e-campus.uab.cat
- Menció realizada: Ingeniería del Software
- Trabajo tutorizado por: Jorge Bernal del Nozal (Ciencias de la Computación)
- Curso 2020/21

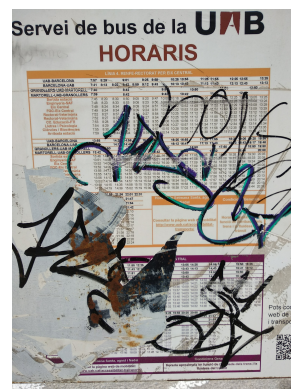


Fig. 1: Panel informativo dañado de una parada de bus

ne. Se conecta a una Web API para consultar los datos para poder consultar una lista de las líneas de buses con su color correspondiente, las paradas que contiene y los horarios de los buses. También tiene estas funcionalidades:

- Permite ver noticias sobre el servicio.
- Muestra las paradas más cercanas a un edificio del campus.
- En la información de cada parada te muestra un mapa con su ubicación, las líneas que pasan.
- Muestra los dos siguientes buses que pasan por una parada de una línea.

Al ser una página web, se obliga al usuario a tener conexión a internet en todo momento para consultar los horarios. Esto se podría evitar si fuese una aplicación móvil instalable que permitiese guardar los horarios e implementar otras funcionalidades específicas de los smartphones.

Para dar solución al problema anterior descrito y mejorar el sistema informático del servicio de buses interno se ha realizado este TFG. Consiste en crear una propuesta de aplicación móvil para usuarios de los buses de UAB. En este informe se muestra el resultado del trabajo realizado. Está estructurado de la forma en que se comienza explicando el estado del arte de las diferentes aplicaciones móviles de transporte público, para luego definir los objetivos y los requisitos. Una vez hecho esto, se muestra las diferentes tecnologías que se han barajado para el desarrollo de la aplicación y cuáles se han escogido. Seguidamente se han identificado los riesgos que podrían afectar al proyecto. Se ha explicado la metodología que se ha seguido, las herramientas utilizadas a la hora de realizar este proyecto y la planificación inicial. Y finalmente se ha expuesto como se ha desarrollado el proyecto, los resultados obtenidos y las conclusiones finales.

2 ESTADO DEL ARTE

Hoy en día las funcionalidades que ofrece el sistema actual puede ser que no sean suficientes para satisfacer las necesidades de los usuarios habituales, porque están acostumbrados a otras aplicaciones de transporte público que ofrecen más funcionalidades.

Ya que estas aplicaciones de transporte son similares a la aplicación que queremos realizar, hay que mirar que hacen bien y que hacen mal para tenerlo en cuenta a la hora de diseñarse la nueva aplicación.

Se ha acotado el estudio a solo las aplicaciones más relevantes para los usuarios del bus de la UAB y se ha creado la Tabla 1 para comparar sus funcionalidades.

De aplicaciones de buses oficiales de otras localidades se ha escogido TMB [2] (Barcelona), TUS [3] (Sabadell) y TMESA [4] (Terrassa) por ser localidades de la que proceden bastantes estudiantes de la UAB. Es posible que los usuarios de los buses internos de la UAB también lo sean de alguno de los dos servicios de transporte ferroviario que tienen parada en la UAB o cerca. Por tanto, se ha añadido FGC [5] y Rodalies de Catalunya [6]. Finalmente también se han añadido al estudio las aplicaciones de terceros Citymapper [7], Moovit [8] y Google Maps [9], que pueden ser utilizadas para consultar información del transporte público.

3 OBJETIVOS

El objetivo de la aplicación es que la puedan utilizar tanto nuevos usuarios como habituales. También se quiere diseñar el sistema de manera que se pueda reutilizar en otros servicios de bus de otras localidades o en otros servicios de transporte público, como por ejemplo el tren.

En un bus cuando se va de pie normalmente hay que agarrarse con una mano para no perder el equilibrio a causa de los frenazos bruscos. Por eso el diseño de la interfaz ha de estar adaptado al uso de una mano. Para conseguir esto

Aplicaciones	Funcionalidades				
	Horarios offline	Bottom Nav Bar	Mapa de paradas	Trayectos favoritos	Paradas favoritas
TMB	×	✓	×	✓*	✓*
TUS / TMESA	×	×	×	×	×
FGC Oficial	×	✓	×	×	×
Rodalies de Catalunya	×	✓	✓	✓	✓
Citymapper	×	×	✓	✓	✓
Moovit	×	✓	✓	×**	✓
Google Maps	×	✓	×	×	✓

* Requiere una cuenta de usuario de TMB.

** Pero se puede guardar una línea en favoritos.

TABLA 1: COMPARATIVA DE FUNCIONALIDADES

hay que tener en cuenta las zonas a las que puede acceder el dedo pulgar en las pantallas cada vez más grandes de los smartphones, como se muestra en la Figura 2.

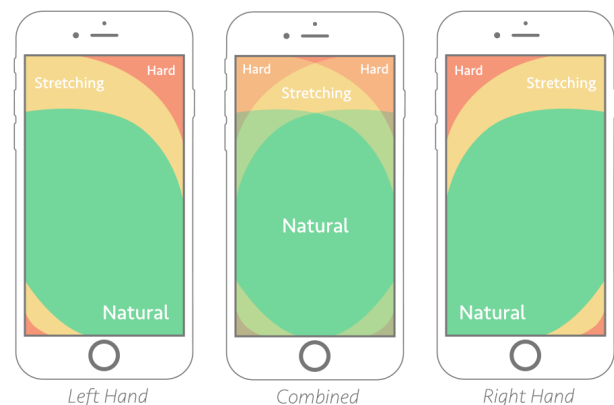


Fig. 2: Mapeo de la zona del pulgar [10]

3.1. Requisitos no funcionales

- RNF-1 La app móvil ha de poder mostrar los horarios sin necesidad de internet.
- RNF-2 La app móvil ha de poder utilizarse como mínimo en móviles Android.
- RNF-3 La app móvil ha de tener una interfaz gráfica con estilo consistente.
- RNF-4 La app móvil ha de poder verse en el idioma catalán, español e inglés.

3.2. Requisitos funcionales

- RF-1 La app móvil ha de mostrar la lista de líneas de bus.
- RF-2 La app móvil ha de mostrar las paradas que tiene una línea de bus.
- RF-3 La app móvil ha de mostrar las horas a las que pasa un bus en una parada.
- RF-4 La app móvil ha de mostrar las líneas que pasan por una parada.

- RF-5 La app móvil ha de mostrar la hora a la que pasa el siguiente bus en una parada.
- RF-6 La app móvil ha de mostrar la lista de paradas que hay en un trayecto entre dos paradas.
- RF-7 La app móvil ha de mostrar la hora de llegada dado un trayecto.
- RF-8 La app móvil ha de mostrar en un mapa todas las paradas.
- RF-9 La app móvil ha de mostrar el trayecto de una línea en un mapa.
- RF-10 La app móvil ha de poder permitir al usuario guardar un trayecto entre dos paradas en favoritos.
- RF-11 La app móvil ha de mostrar en la pantalla principal la hora de llegada del siguiente bus de los trayectos favoritos.
- RF-12 La app móvil ha de poder permitir cambiar el idioma.
- RF-13 El sistema ha de permitir a los administradores actualizar horarios, paradas y líneas.

En la Figura 3 se representa estas funcionalidades en un diagrama de casos de uso.



Fig. 3: Diagrama de casos de uso de la aplicación móvil

4 TECNOLOGÍAS

Para realizar el proyecto se han barajado la elección de las siguientes tecnologías, donde se han tenido en cuenta la documentación, comunidad y facilidad de uso.

4.1. Aplicación móvil

- **Web:** Aunque el sistema actual consiste en una página web, tiene ciertas limitaciones frente a una aplicación nativa.
- **PWA [11]:** Añade algunas funcionalidades nativas a una página web como acceso offline o notificaciones.
- **Nativo:** La opción más óptima es realizar un desarrollo nativo utilizando las tecnologías de cada sistema operativos, pero conlleva más costes de desarrollo.

- **Flutter [12]:** Aún siendo un framework para desarrollar aplicaciones multiplataforma relativamente nuevo, está teniendo un crecimiento muy acelerado gracias al apoyo de Google y su gran comunidad. Permite tener la misma interfaz y funcionalidad en todas las plataformas utilizando el mismo código.

Se ha escogido Flutter como tecnología de desarrollo móvil por su baja curva de aprendizaje y la rapidez en la que se puede construir interfaces modulares siguiendo Material Design [13] o un diseño propio.

4.2. Base de datos

- **MariaDB [14]:** Es una base de datos de código abierto derivada de MySQL lo que le otorga plena compatibilidad con sistemas que trabajen con esta.
- **Couchbase [15]:** Es una base de datos NoSQL de código abierto. Su modelo está orientado a documentos. Permite la sincronización con Couchbase Lite [16], la base de datos pensada para implementarse en dispositivos.
- **Cloud Firestore [17]:** Es la base de datos NoSQL de licencia privativa, ya integrada en el servicio Firebase [18] de Google. También ofrece otras funcionalidades que pueden ser útiles para este proyecto. Solo se ofrece como servicio en su Cloud y dispone de un plan gratuito limitado. Es necesario un SDK específico para cada plataforma que se quiera comunicar con el servicio.

Se ha optado por utilizar Cloud Firestore porque ya dispone de alojamiento Cloud y por relacionarse bien con Flutter, al ser las dos tecnologías de Google.

5 RIESGOS

Riesgo: Planificación optimista

Efecto: Retardo en la finalización de las tareas

Impacto: Crítico

Solución: Replanificar las tareas

Riesgo: Imposibilidad de implementar en Flutter el mapa con las paradas.

Efecto: No se puede acceder a un mapa con las paradas.

Impacto: Medio

Solución: Insertar una fotografía ampliable del mapa de paradas.

Riesgo: Firebase deja de tener un plan gratuito

Efecto: Hay que pagar lo que cueste o encontrar otra alternativa, con el consiguiente coste de tiempo en rediseño del sistema.

Impacto: Crítico

Solución: Rediseñar el sistema para utilizar alguna de las alternativas barajadas.

Riesgo: No se tiene acceso a los archivos de los horarios de los buses.

Efecto: Hay que utilizar otras alternativas par obtener los datos que se insertaran en el sistema, con la posibilidad de retrasar la planificación

Impacto: Medio

Solución: Exportarlos haciendo llamadas a la Web API existente o insertarlos a mano.

6 METODOLOGÍA

Al haber mucha incertidumbre por los problemas que puedan surgir a la hora de realizar el desarrollo del proyecto, se ha escogido Extreme Programming [19] como metodología a seguir. Esto implica un desarrollo iterativo e incremental tanto en la realización de la aplicación como en la redacción de la documentación. Para validar que se cumplen los requisitos se han realizado pruebas. Se ha perseguido la simplicidad del código y que esté autodocumentado para facilitar el mantenimiento.

7 HERRAMIENTAS

Para la realización de este proyecto se han utilizado las siguientes herramientas:

7.1. Software:

- **PlantUML [20]:** Con ella se ha creado el diagrama de casos de uso.
- **LaTeX [21]:** Se ha utilizado para componer este informe.
- **Git [22] y GitHub [23]:** Utilizado para tener un control de versiones, copia de seguridad del proyecto.
- **Visual Studio Code [24]:** Para programar la aplicación en Flutter y utilizar las tres herramientas anteriores.
- **Microsoft PowerPoint [25]:** Usado para realizar los bocetos del diseño de las pantallas de la aplicación.

7.2. Hardware:

- **Ordenador con Windows 10:** Se recomienda una configuración mínima de 1.6 GHz de CPU [26], 2 GB de espacio en disco y 2 GB de RAM [27]. Todas las herramientas de software utilizadas están también disponibles para MacOS y Linux.

8 PLANIFICACIÓN

Se ha establecido la siguiente planificación del proyecto:

21 feb. - 15 mar. : Documentar en el Informe inicial:

- Establecer objetivos
- Elegir tecnologías
- Identificar riesgos
- Establecer metodología de trabajo.
- Analizar el estado del arte
- Planificar proyecto

Tener un primer contacto con Flutter y Firebase haciendo una pequeña aplicación.

15 mar. - 25 abr. : Desarrollar la aplicación móvil con las funcionalidades básicas: cambiar idioma, ver lista de líneas, paradas, siguiente bus y horarios. Hacer la web de administración para poder importar los datos de la base de datos. Documentar informe de progreso I.

26 abr. - 30 may. : Desarrollar las funcionalidades avanzadas: generar trayectos, trayectos favoritos, ver información detallada de la parada. Documentar informe de progreso II.

31 may. - 20 jun. : Desarrollar las funcionalidades del mapa y alertas de servicio. Testear que la aplicación cumple todos los requisitos. Documentar el informe final.

21 jun. - 27 jun. : Preparar presentación y hacer dossier del TFG.

28 jun. - 4 jul. : Hacer póster.

9 DESARROLLO

9.1. Base de datos

Se ha diseñado un archivo JSON que contiene toda la información que contendrá la base de datos. Este tiene los datos normalizados para facilitar por si se tiene que modificar manualmente. Esto ha permitido añadir las coordenadas donde se encuentra cada parada fácilmente. Para la generación de este archivo JSON se ha creado un programa en Dart que ha utilizado los datos obtenidos con la API actual para generar el archivo.

Ejemplo simplificado de archivo JSON

```
{
  "lines": {
    "1": {
      "name": "RENFE-RECTORAT",
      "code": "L1",
      "color": "#0000FF"
    }
  },
  "stops": {
    "1": {
      "name": "Eureka a Vila"
    }
  },
  "lines_schedule": {
    "1": [
      {
        "id": "21",
        "schedule": [
          "07:45",
          "08:20"
        ]
      }
    ]
  }
}
```

Para actualizar los datos de la aplicación se ha creado una página web con Flutter. Está formada por una pantalla de inicio de sesión que restringe la entrada a solo los usuarios dados de alta en Firebase Authentication [28]. Una vez validado el inicio de sesión, el usuario accede a un menú donde tiene la posibilidad de ir a la página de importación de datos o de enviar alertas. En la página de importación de datos se muestra un botón para subir el archivo JSON y otro botón para que la web la información desnormalizada en la base de datos. La conversión se hace en el lado del cliente, esto permite que la web se pueda alojar en cualquier servicio de alojamiento de páginas webs estáticas. Los datos se desnormalizan para utilizar las llamadas justas y necesarias por parte de los clientes, ya que se almacenan en una base de datos NoSQL que cobra por llamada. A continuación se muestra la estructura de datos que genera la web antes de subir los datos a Firestore, donde lines, stops y schedules corresponderían a los datos que se suben en cada colección de la base de datos.

Ejemplo en la base de datos

```

"lines": {
  "1": {
    "name": "RENFE-RECTORAT",
    "code": "L1",
    "color": "#0000FF"
  }
},
"stops": {
  "1": {
    "name": "Eureka a Vila",
    "location": "41.50290715447545,
      2.1100370139549036"
    "lines": [
      {
        "id": "1",
        "name": "RENFE-RECTORAT",
        "code": "L1",
        "color": "#0000FF"
        "schedule": [
          "07:45",
          "08:20"
        ]
      }
    ]
  }
},
"schedules": {
  "1": {
    "name": "RENFE-RECTORAT",
    "code": "L1",
    "color": "#0000FF",
    "stops": [
      {
        "id": "21",
        "name": "Estació de Renfe",
        "location": "41.50290715447545,
          2.1100370139549036",
        "schedule": [
          "07:45",
          "08:20"
        ]
      }
    ]
  }
}

```

También se ha realizado otra página de administración con un formulario para enviar alertas de servicio que consisten en un título, un texto, y fecha y hora a la que se ha enviado. Una vez enviada se guarda en colección de alertas en la base de datos.

9.2. Inicialización proyecto Flutter

Para creación de la aplicación se ha utilizado la versión 2.2.0 de Flutter, la última versión estable disponible en el momento de desarrollo. Todos los paquetes externos utilizados se han escogido teniendo en cuenta que deben permitir la exportación para Android y Web. Se ha activado Null safety [29] en el proyecto para restringir que todas las variables no pueden ser nulas, a no ser que se especifique. Para seguir un estilo de código y detectar malas prácticas se ha utilizado el paquete `very_good_analysis` [30]. Se ha subido el código a un repositorio público en GitHub para tener un control de versiones con copia de seguridad, y para automatizar la exportación Web y publicación en Github Pages [31] de la aplicación, que la realiza cada vez se publica un commit.

9.3. Internacionalización y localización

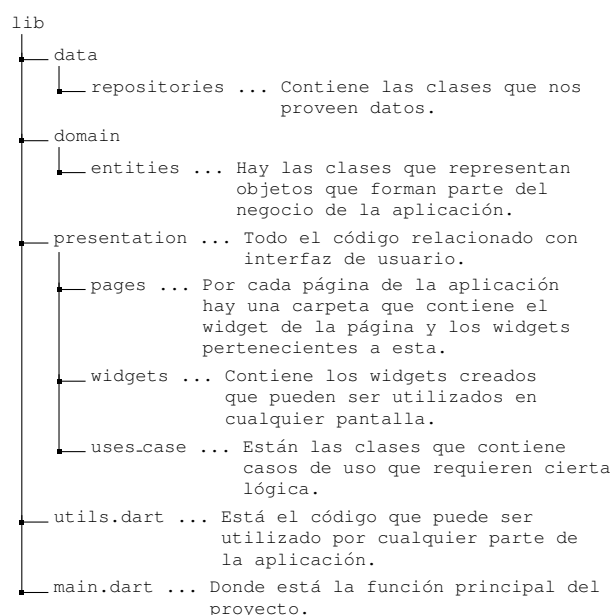
En vez de utilizar el sistema de internacionalización y localización propuesto por Flutter, se ha utilizado el paquete `easy_localization` [32], y

`easy_localization_loader` [33] para cargar los archivos de localizaciones, ya que permiten diferentes tipos de formato a la hora de leer los archivos de localizaciones. Para localizar los nombres de las localizaciones a la hora de seleccionarlás en la app, se ha utilizado el paquete `flutter_localized_locales` [34]. Para editar los archivos de localizaciones se ha utilizado la extensión para VSCode llamada `i18n Ally` [35] porque soporta diferentes tipos de archivos de localización.

Se ha decidido utilizar este sistema porque el sistema oficial de Flutter nos obliga a trabajar con archivos de localización ARB, que está poco soportado por herramientas de edición de localizaciones gratuitas, ya que solo se utiliza en Flutter. Como alternativa se ha utilizado archivos de localización en formato YAML porque es fácilmente editable manualmente, y si se necesitase se puede convertir fácilmente a JSON.

9.4. Arquitectura de la app

En el diseño de la estructura del código de la aplicación se ha basado en las recomendaciones de Clean Architecture. Como resultado de esto se ha organizado el código Dart contenido en la carpeta `lib` en la siguiente estructura de subcarpetas:



9.5. Pantallas de la Aplicación

Par que se pueda utilizar con una mano la aplicación se ha creado un Bottom Navigation Bar [36] para cambiar entre sus páginas principales: Inicio, Líneas, Alertas, Mapa y Trayecto.

9.5.1. Tema

Se ha utilizado la solución que nos provee Flutter para implementar temas que estén accesibles y se apliquen en toda la aplicación. Esto permite que toda la aplicación tenga el mismo estilo y se pueda cambiar fácilmente. Para facilitar la construcción de nuevas pantallas que mantengan el mismo estilo se ha creado widgets que extienden de widgets

por defecto pero se les aplica una personalización. Como los colores de las líneas eran demasiado claros y saturados, cada vez que se muestran por pantalla pasan por un filtro que les reduce el brillo y la saturación para que se puedan mostrar los iconos de las líneas correctamente formados por su letra y número en blanco sobre su color correspondiente.

9.5.2. Página de las líneas

La primera página que se ha implementado ha sido la página de Líneas que, tras una consulta a la base de datos, muestra en una lista las líneas de bus representadas por su nombre, su código y su color (Figura 4).

Al clicar a una línea, se vuelve a consultar a la base de datos para obtener las paradas y horarios de esa línea, y se abre otra página donde se muestra la página de la línea.

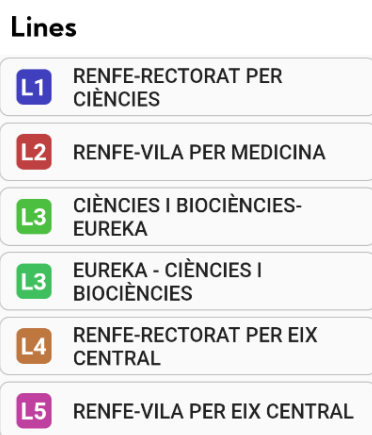


Fig. 4: Página donde se muestra el listado de líneas de bus

9.5.3. Página de detalles de una línea

En esta página se muestra las paradas de la línea de bus en una línea de tiempo vertical donde por cada parada se muestra el nombre y los dos siguientes horas a las que pasará el bus de esa línea por ella (Figura 5).



Fig. 5: Página donde se muestra la información de una línea

9.5.4. Página de detalles de una parada

En esta página se muestra la información relacionada con una parada (Figura 6). Contiene un mini mapa en el que se muestra donde está situada la parada. Se muestra una lista de las líneas que pasan por esa parada reutilizando un widget creado para la página de mostrar las líneas. Si se presiona un de estas líneas te lleva a una pantalla donde se muestra todas las horas que pasa el bus de esa línea por esa parada (Figura 7).



Fig. 6: Página donde se muestra la información de una línea

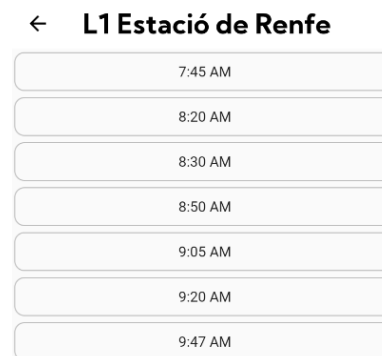


Fig. 7: Página donde se muestra el horario de un bus de una línea en una parada

9.5.5. Página de configuración

Se ha creado una pantalla para cambiar las configuraciones de la aplicación (Figura 8). Esta es accesible a través de un icono en la pantalla de inicio. Aunque la aplicación se localiza según el idioma del dispositivo, también en esta pantalla se puede cambiar el idioma de la aplicación a través de un desplegable donde los idiomas a escoger salen traducidos al idioma del usuario. Cuando se cambia el idioma se actualiza la interfaz automáticamente para hacer el cambio de idioma.



Fig. 8: Página de configuración de la aplicación

9.5.6. Página de trayecto

En esta página se muestran tres desplegables. En el primer desplegable se escoge la línea, y en los dos siguientes la parada de salida y de llegada. Una vez escogida la línea se habilita el segundo desplegable para escoger la parada de salida de esa línea. Para evitar que el usuario seleccione una parada anterior a la de la salida, el desplegable de la parada de llegada se habilita cuando el usuario escoge la de salida. Y cuando se habilita este, solo se muestran paradas posteriores a la de salida para evitar que el usuario ponga una parada anterior a la de salida. Una vez el usuario ha rellenado esos tres campos, automáticamente le aparecerá en la misma página una representación del trayecto escogido en una línea del tiempo vertical donde se muestran los nombres de las paradas y la hora a la que pasara el usuario si escoge ese trayecto ahora.

También en esta página se muestra un selector de la hora de salida que por defecto muestra que está seleccionada la hora actual. Flutter no proporciona ningún widget que sea un campo para introducir y mostrar una hora, solo nos proporciona una función que llama a una ventana para escoger una hora. Por eso se ha creado el widget `time_picker`, tomando como referencia las funciones de código del widget desplegable de Flutter para que a la hora de implementarlo sea similar.

Una vez se ha escogido un trayecto, el usuario puede guardarlo o eliminarlo de los trayectos favoritos presionando el botón de arriba a la derecha. Esta acción hará que se añada o se elimine inmediatamente el trayecto de la página del inicio gracias a la utilización del paquete `provider`, que permite notificar la propagación de cambios en la interfaz.

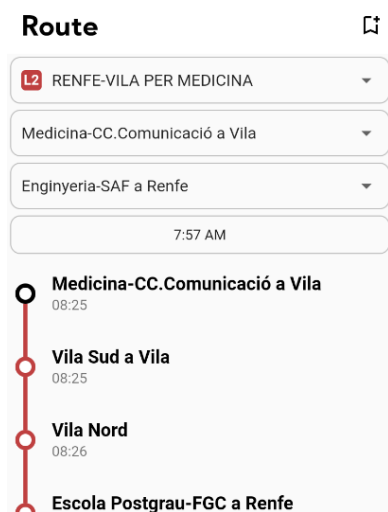


Fig. 9: Página donde se puede buscar, mostrar y guardar un trayecto

9.5.7. Página de inicio

La página de inicio (Figura 10) será la primera página que el usuario verá al entrar a la aplicación. Por lo tanto, tiene que mostrar la información que más valor le aporte al usuario. Una información importante que valora muchísimo el usuario es la hora a la que pasa el siguiente bus en sus trayectos favoritos tan pronto abra la aplicación. Para

mostrar esa información condensada en un widget se ha diseñado el widget `route_tile`. Este muestra las paradas de salida y llegada con las horas a las el bus estaría en esas paradas. También se ha desarrollado un widget llamado `countdown_timer`. A la derecha muestra el tiempo simplificado restante, que se colorea en naranja cuando quedan 5 min para alertar al usuario, como se puede apreciar en la Figura 11. Además, en este widget se muestra una pequeña cadena horizontal de iconos que representan las líneas que coge el usuario en ese trayecto. Este widget se actualiza cuando el tiempo restante es -1 segundo para mostrar siempre el trayecto siguiente. Si el usuario presiona un trayecto favorito se le abre la página donde se selecciona el trayecto con los campos rellenos. De esta manera el usuario puede visualizar el trayecto con más detalle y buscar otro cambiando cualquier campo.

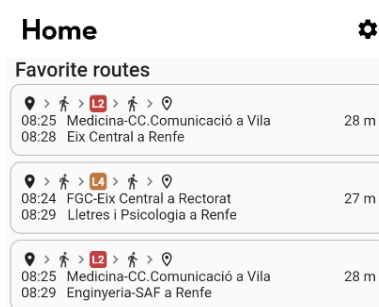


Fig. 10: Página principal de la aplicación



Fig. 11: Widget `route_tile` con el `countdown_timer` en naranja cuando queda poco tiempo para que pase el bus

9.5.8. Página de mapa de paradas

Para realizar el mapa donde se muestran las paradas se han barajado la utilización de uno de estos tres paquetes de mapas para Flutter: `flutter_map` [37], `mapbox_gl` [38], `google_maps_flutter` [39]. Se ha escogido `flutter_map` porque las características que lo destacan sobre los demás es que permite trabajar con widgets de Flutter dentro del mapa, su facilidad de implementación y no depende de un servicio de mapas que restrinja su acceso con una API key.

En la pantalla de mapas de paradas se muestra estas en un mapa representadas por una señal negra con el borde blanco para verse bien sobre cualquier superficie del mapa. Al presionar una parada se muestra un pop-up que muestra el nombre de la parada y las líneas que pasan por ella. Si se presiona el pop-up se abre la página de detalles de la parada seleccionada. También esta pantalla permite seleccionar una línea en un desplegable para mostrar su recorrido en el mapa, como se observa en Figura 12.



Fig. 12: Página donde se muestra el mapa con las paradas

9.5.9. Página de alertas

Se ha realizado una página donde se visualizan las alertas enviadas por la administración (Figura 13). Por cada alerta se muestra el título, el contenido, y la fecha y hora a la que se ha publicado. La fecha y hora se adapta al huso horario que tiene el usuario en el móvil. También se muestra la fecha en el formato e idioma que está seleccionado en la aplicación.

Alertes

Lorem ipsum dolor sit amet
13 de juny de 2021 21:26
consectetur adipiscing elit, sed eiusmod tempor
incididunt ut labore et dolore magna aliqua.
Ut enim ad minim veniam, quis nostrud exercitation
ullamco laboris nisi ut aliquid ex ea commodo
consequat

Modificacions horaris de busos a partir del 4 de
febrer
13 de juny de 2021 21:22
Lorem ipsum dolor sit amet, consectetur adipiscing
elit, sed eiusmod tempor incididunt ut labore et dolore
magna aliqua. Ut enim ad minim veniam, quis nostrud
exercitation ullamco laboris nisi ut aliquid ex ea
commodi consequat.

Fig. 13: Página donde se muestran las alertas del servicio

9.6. Datos de la Aplicación

9.6.1. Trayecto

Un trayecto se guarda en la clase Route que está formada por clases RouteLeg. Cada RouteLeg representa los tramos donde el usuario viaja en una línea. Un RouteLeg está compuesto por la Línea y los Steps. Cada Steps representa las paradas de ese trayecto, y contiene el nombre de la parada y la hora a la que pasaría el usuario en ese trayecto. Esta estructura de datos permite implementar trayectos en los que implique hacer trasbordo, aunque en este proyecto no se ha llegado a implantar.

9.6.2. Base de datos de trayectos favoritos

Para guardar los trayectos favoritos en el dispositivo se ha utilizado la base de datos clave-valor HiveDB [40] exclusiva para Dart. Esta base de datos permite ser llamada con código Dart que compile para cualquier plataforma soportada. Por cada trayecto favorito se guarda el identificador de la línea, el identificador de la parada salida y el identificador de la parada de destino. Para saber si un trayecto existe

ya en la base de datos o para eliminarlo, se ha tenido que establecer que el índice con el que se identifica el trayecto favorito en la base de datos sea el resultado de aplicar un hash a los tres campos que lo conforman.

10 PRUEBAS

A medida que se iba desarrollando la aplicación se han ido probando manualmente sus funcionalidades y validándolas comparando los resultados con la aplicación anterior, ya que las dos tienen la misma fuente de datos.

Para asegurar el buen funcionamiento de la lógica de la aplicación en sus partes crítica se han hecho una batería de pruebas unitarias. En las partes de la aplicación que se quería testear, pero no se podían hacer a través de código, se han diseñado casos de prueba.

11 RESULTADO

Como resultado de este proyecto, se ha obtenido una aplicación móvil funcional que puede substituir a la antigua web de consulta de información de los buses internos de la UAB, ya que implementa prácticamente todas las funcionalidades de esta. Además, añade funcionalidades presentes en otras aplicaciones similares, como por ejemplo la posibilidad de consultar un trayecto y guardarlo en favoritos para tenerlos accesibles. En el siguiente video se muestra el funcionamiento de la aplicación:

Vídeo de la aplicación móvil:

<https://youtu.be/iaHwSxgoxMI>

Aunque la aplicación ha sido desarrollada pensando en ser instalada en un móvil, gracias al framework utilizado se ha podido también crear una aplicación web con las mismas funcionalidades.

12 CONCLUSIONES

El resultado del proyecto ha sido satisfactorio. Aunque en el periodo de desarrollo el progreso iba con un poco de retraso respecto a la planificación inicial se ha conseguido cumplir todos los requisitos del proyecto a tiempo.

12.1. Dificultades

A pesar de que Flutter tiene una buena documentación, no se puede decir lo mismo de muchos paquetes necesarios para realizar una aplicación de calidad. Esto ha hecho que para encontrar la solución a algunos problemas tuvieses que buscar en las Issues de los repositorios de estos paquetes o en los pocos ejemplos que proporcionan. Otra desventaja del framework ha sido que por culpa de su juventud y su constante evolución se han encontrado paquetes con poco mantenimiento que no se han adaptado a tiempo a cambios del framework, como por ejemplo la utilización de null safety. Este problema se ha encontrado cuando se estaba probando entre diferentes paquetes de mapas se tuviese que utilizar forks o arreglar algún bug de estos.

A pesar de que la exportación a web de la aplicación funciona perfectamente en local, a la hora de publicar la aplica-

ción web en Github Pages los mapas no se muestran porque el servidor tiene CORS [41] activado y no hay forma de configurarlo o desactivarlo.

12.2. Aprendizaje

Gracias a este TFG he podido aprender el framework Flutter, que no había visto cursando en el grado de Ingeniería Informática. Al trabajar con él he adquirido una buena base de conocimientos sobre este y me ha despertado el interés para utilizarlo en un futuro para realizar otras aplicaciones móviles, a pesar de los problemas antes mencionados. Como para la base de datos de la aplicación se ha utilizado una Firestore he tenido que aprender a modelar los datos teniendo en cuenta en que en esta no existen joins como en SQL, haciendo que cada vez que se requiera una nueva consulta compuesta se tuviese que modelar los datos para evitar más consultas de las necesarias. Me ha gustado aplicar los conocimientos aprendidos en asignaturas del grado, como por ejemplo Diseny del Software, Requisites del Software y Test i Qualitat del Software.

12.3. Trabajo futuro

Ya que Flutter también permite la exportación para dispositivos iOS se podría intentar exportar a esta plataforma si se dispone de un Mac. Aprovechando que es una aplicación móvil se podría añadir funcionalidades como el uso de la cámara para escanear QR en una parada o el uso del GPS del móvil para obtener la parada más cercana. Estas se podrían haber hecho si se hubiese dispuesto de más tiempo. Como trabajo futuro se podría adaptar la aplicación a buses de otras poblaciones, ya que la interfaz y experiencia de usuario conseguida mejora en algunos puntos a las aplicaciones de estas.

AGRADECIMIENTOS

Quisiera agradecer a Jorge Bernal por ser el tutor de este TFG, y darme el soporte y consejos en todo momento para realizarlo. También agradezco a mi familia todo el apoyo que me han proporcionado y por haber hecho de beta testers de la aplicación.

REFERENCIAS

- [1] “BUSOS UAB.” (), dirección: <http://appbuses.accessibilitat-transports.uab.cat/> (visitado 07-03-2021).
- [2] “TMB.” (), dirección: <http://appbuses.accessibilitat-transports.uab.cat/> (visitado 07-03-2021).
- [3] “TUS-Bus Sabadell.” (), dirección: <https://play.google.com/store/apps/details?id=com.geoactio.tus> (visitado 07-03-2021).
- [4] “TMESA - Bus Terrassa.” (), dirección: <https://play.google.com/store/apps/details?id=com.geoactio.tmesa> (visitado 07-03-2021).
- [5] “app — Ferrocarrils de la Generalitat de Catalunya.” (2020), dirección: <https://www.fgc.cat/es/app/> (visitado 13-03-2021).
- [6] “App Rodalies de Catalunya.” (), dirección: http://rodalies.gencat.cat/es/atencio_al_client/app_rodalies/ (visitado 13-03-2021).
- [7] “Citymapper - Rutas en transporte público.” (), dirección: <https://play.google.com/store/apps/details?id=com.citymapper.app.release&hl=es> (visitado 13-03-2021).
- [8] “Moovit: Horarios de Tren, Metro y Bus.” (), dirección: <https://play.google.com/store/apps/details?id=com.tranzmate&hl=es&gl=US> (visitado 13-03-2021).
- [9] “Google Maps - Navegación y transporte público.” (), dirección: <https://play.google.com/store/apps/details?id=com.google.android.apps.maps&hl=es&gl=US> (visitado 13-03-2021).
- [10] S. Ingram. “The Thumb Zone: Designing For Mobile Users — Smashing Magazine.” (sep. de 2016), dirección: <https://www.smashingmagazine.com/2016/09/the-thumb-zone-designing-for-mobile-users/> (visitado 07-03-2021).
- [11] Wikipedia. “Aplicación web progresiva — Wikipedia, The Free Encyclopedia.” (2021), dirección: https://es.wikipedia.org/wiki/Aplicaci%C3%B3n_web_progresiva (visitado 07-03-2021).
- [12] “Flutter - Beautiful native apps in record time.” (), dirección: <https://flutter.dev/> (visitado 11-03-2021).
- [13] Google. “Introduction - Material Design.” (), dirección: <https://material.io/design/introduction> (visitado 10-03-2021).
- [14] “MariaDB Foundation - MariaDB.org.” (), dirección: <https://mariadb.org/> (visitado 07-03-2021).
- [15] “Couchbase: Best NoSQL Cloud Database Service.” (), dirección: <https://www.couchbase.com/> (visitado 07-03-2021).
- [16] “Lite — Couchbase.” (), dirección: <https://www.couchbase.com/products/lite> (visitado 07-03-2021).
- [17] “Cloud Firestore — Firebase.” (), dirección: <https://firebase.google.com/docs/firestore?hl=es> (visitado 07-03-2021).
- [18] “Cloud Firestore — Firebase.” (), dirección: <https://firebase.google.com/?hl=es> (visitado 07-03-2021).
- [19] CCIA Universitat d’Alacant. “Programación eXtrema.” (jul. de 2010), dirección: <http://www.jtech.ua.es/j2ee/2006-2007/restringido/met/sesion02-apuntes.html> (visitado 07-03-2021).

- [20] “PlantUML.” (), dirección: <https://plantuml.com/es/> (visitado 10-03-2021).
- [21] “LaTeX – A document preparation system.” (), dirección: <https://www.latex-project.org/> (visitado 10-03-2021).
- [22] “Git.” (), dirección: <https://git-scm.com/> (visitado 10-03-2021).
- [23] “GitHub: Where the world builds software.” (), dirección: <https://github.com/> (visitado 10-03-2021).
- [24] “Visual Studio Code - Code Editing. Redefined.” (), dirección: <https://code.visualstudio.com/> (visitado 10-03-2021).
- [25] “Software de presentación de diapositivas Microsoft PowerPoint — Microsoft 365.” (), dirección: <https://www.microsoft.com/es-es/microsoft-365/powerpoint> (visitado 10-03-2021).
- [26] “Windows install.” (), dirección: <https://flutter.dev/docs/get-started/install/windows> (visitado 07-03-2021).
- [27] “Requirements for Visual Studio Code.” (), dirección: <https://code.visualstudio.com/docs/supporting/requirements> (visitado 07-03-2021).
- [28] “Firebase Authentication — Simple, free multi-platform sign-in.” (), dirección: <https://firebase.google.com/products/auth> (visitado 20-06-2021).
- [29] “Sound null safety — Dart.” (), dirección: <https://dart.dev/null-safety> (visitado 20-06-2021).
- [30] verygood.ventures. “Very Good Analysis: Lint rules for Dart and Flutter used internally at Very Good Ventures.” (), dirección: https://pub.dev/packages/very_good_analysis (visitado 20-06-2021).
- [31] “GitHub Pages — Websites for you and your projects.” (), dirección: <https://pages.github.com/> (visitado 20-06-2021).
- [32] “Easy and Fast internationalizing and localization your Flutter Apps.” (), dirección: https://pub.dev/packages/easy_localization (visitado 20-06-2021).
- [33] “Easy Localization Loader custom assets loaders for easy_localization.” (), dirección: https://pub.dev/packages/easy_localization_loader (visitado 20-06-2021).
- [34] “flutter localized locales: Flutter plugin which provides locale code to name mappings for 563 locales.” (), dirección: https://pub.dev/packages/flutter_localized_locales (visitado 20-06-2021).
- [35] “i18n Ally: All in one i18n extension for VS Code.” (), dirección: <https://github.com/lokalise/i18n-ally> (visitado 20-06-2021).
- [36] “Bottom navigation - Material Design.” (), dirección: <https://material.io/components/bottom-navigation> (visitado 20-06-2021).
- [37] “flutter_map: A Flutter map widget inspired by Leaflet.” (), dirección: https://pub.dev/packages/flutter_map (visitado 20-06-2021).
- [38] “Flutter Mapbox GL: A Mapbox GL flutter package for creating custom maps.” (), dirección: https://pub.dev/packages/mapbox_gl (visitado 20-06-2021).
- [39] “Google Maps for Flutter: A Flutter plugin for integrating Google Maps in iOS and Android applications.” (), dirección: https://pub.dev/packages/google_maps_flutter (visitado 20-06-2021).
- [40] “Hive: Lightweight and blazing fast key-value database written in pure Dart.” (), dirección: <https://github.com/hivedb/hive> (visitado 20-06-2021).
- [41] Wikipedia. “Intercambio de recursos de origen cruzado.” (2020), dirección: https://es.wikipedia.org/wiki/Intercambio_de_recursos_de_origen_cruzado (visitado 20-06-2021).

APÉNDICE

A.1. Ejemplo de algunos casos de prueba realizados

Test Case ID	TC 1.2		
Description	Ir a la página Trayecto y generar una trayecto que empiece a partir de una hora determinada		
Module	UAB Bus App		
Prepared By	Sergi Luque	Date Prepared	14/06/2021
Reviewed / Updated		Date Reviewed	14/06/2021
Tested By	Sergi Luque	Date Tested	14/06/2021
Test Activities			
Sl. No.	Step Description	Expected Results	
1	Ir a la página Trayecto.	Se muestra la página con un desplegable habilitado y dos inhabilitados, y un botón que muestra la palabra 'Ahora'.	
2	Seleccionar en el primer desplegable la línea [línea].	Se muestra el icono y nombre de la línea [línea] en el primer desplegable y se habilita el segundo desplegable con las paradas de la línea [línea].	
3	Seleccionar en el segundo desplegable la parada [parada1].	Se muestra el nombre de la parada [parada1] en el segundo desplegable y se habilita el tercer desplegable con paradas posteriores a la parada [parada1].	
4	Seleccionar en el segundo desplegable la parada [parada2].	Se muestra el trayecto de las paradas que hay entre las paradas [parada1] y [parada2] en la línea [línea] y las horas a las que pasaría un usuario por cada parada si comenzase ese trayecto a partir de la hora actual. También se muestra el botón para guardar o eliminar el trayecto.	
5	Presionar el botón 'Ahora' y seleccionar la hora [hora].	Se muestra el mismo trayecto que antes, pero ahora las horas a las que pasaría un usuario por cada parada son si comenzase ese trayecto a partir de la hora [hora].	
Test Data Sets			
Data Type	Data Set 1	Data Set 2	Data Set 3
línea	L1 RENFE-RECTORAT PER CIENCIES	L2 RENFE-VILA PER MEDICINA	L5 RENFE-VILA PER EIX CENTRAL
parada1	Estació de Renfe	Vila Nord	Vila Nord
parada2	Ciències i Biociències a Rectorat	Eix Central a Renfe	Eureka Renfe
hora	10:00	0:00	23:55
Test Case Result		Successful	

Test Case ID	TC 2.1		
Description	Generar un trayecto, guardarlo y visualizarlo en la pantalla de inicio		
Precondition	No tiene que haber ningún trayecto guardado en favoritos.		
Module	UAB Bus App		
Prepared By	Sergi Luque	Date Prepared	14/06/2021
Reviewed / Updated		Date Reviewed	14/06/2021
Tested By	Sergi Luque	Date Tested	14/06/2021
Test Activities			
Sl. No.	Step Description	Expected Results	
1	Ir a la página Inicio.	La sección "Trayectos favoritos" está vacía.	
2	Ir a la página Trayecto.	Se muestra la página con un desplegable habilitado y dos inhabilitados, y un botón que muestra la palabra 'Ahora'.	
3	Seleccionar en el primer desplegable la línea [línea].	Se muestra el icono y nombre de la línea [línea] en el primer desplegable y se habilita el segundo desplegable con las paradas de la línea [línea].	
4	Seleccionar en el segundo desplegable la parada [parada1].	Se muestra el nombre de la parada [parada1] en el segundo desplegable y se habilita el tercer desplegable con paradas posteriores a la parada [parada1].	
5	Seleccionar en el segundo desplegable la parada [parada2].	Se muestra el trayecto de las paradas que hay entre las paradas [parada1] y [parada2] en la línea [línea] y las horas a las que pasaría un usuario por cada parada si comenzase ese trayecto a partir de la hora actual. También se muestra el botón para guardar representado por la figura del contorno de un marcapáginas con el símbolo "+".	
6	Presionar el botón de guardar trayecto en favoritos.	La figura del botón de guardar en favoritos ha cambiado a la figurada de un marcapáginas todo negro sin el símbolo "+".	
7	Ir a la página Inicio.	Se muestra la información del trayecto entre la parada [parada1] y [parada2] en la línea [línea].	
Test Data Sets			
Data Type	Data Set 1	Data Set 2	Data Set 3
línea	L1 RENFE-RECTORAT PER CIENCIES	L2 RENFE-VILA PER MEDICINA	L5 RENFE-VILA PER EIX CENTRAL
parada1	Estació de Renfe	Vila Nord	Vila Nord
parada2	Ciències i Biociències a Rectorat	Eix Central a Renfe	Eureka Renfe
Test Case Result		Successful	

Test Case ID		TC_2.2	
Description		Eliminar un trayecto ya guardado en favoritos.	
Precondition		Tiene que haber guardado en favoritos un trayecto entre las paradas [parada1] y [parada2] en la línea [línea] .	
Module		UAB Bus App	
Prepared By	Sergi Luque	Date Prepared	14/06/2021
Reviewed / Updated		Date Reviewed	14/06/2021
Tested By	Sergi Luque	Date Tested	14/06/2021
Test Activities			
Sl. No.	Step Description	Expected Results	
1	Ir a la página Inicio.	La sección "Trayectos favoritos" contiene un trayecto entre las paradas [parada1] y [parada2] en la línea [línea].	
2	Presionar el trayecto favorito.	La aplicación te lleva a la página Trayecto con los campos rellenos con la información correspondiente al trayecto guardado. El botón de eliminar de los favoritos tiene la forma de un marcapáginas negro.	
6	Presionar el botón de eliminar trayecto.	La figura del botón de eliminar de los favoritos ha cambiado a la figura del contorno de un marcapáginas con el símbolo "+".	
7	Ir a la página Inicio.	Ya no se muestra el trayecto entre las paradas [parada1] y [parada2] de la línea [línea].	
Test Data Sets			
Data Type	Data Set 1	Data Set 2	Data Set 3
línea	L1 RENFE-RECTORAT PER CIENCIES	L2 RENFE-VILA PER MEDICINA	L5 RENFE-VILA PER EIX CENTRAL
parada1	Estació de Renfe	Vila Nord	Vila Nord
parada2	Ciències i Biociències a Rectorat	Eix Central a Renfe	Eureka Renfe
Test Case Result		Successful	

A.2. Bocetos de las pantallas de la aplicación

Home

Favorite routes

Start stop name - Next bus time
End stop name - Time of arrival

Configuration

English

Route

Start stop name

End stop name

Time

Confirm

Lines

L1 Line name

L2 Line name

Line name

Stop name 10:00

Stop name 10:05

Stop name

L1 Line name

L4 Line name

L4 Stop name

10:00

10:05

10:10

Alerts

Title 10/10/21
Lorem ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries

Title 10/10/21
Lorem ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries

A.3. Pantallas de la web de administración



Fig. 14: Página de inicio de sesión

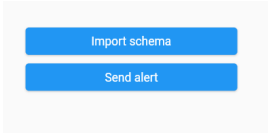


Fig. 15: Página de menú

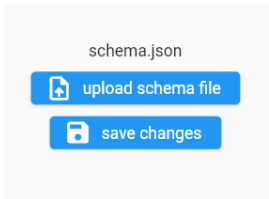


Fig. 16: Página de importación del los datos

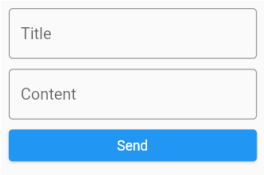


Fig. 17: Página de formulario de envío de alertas